Graph Matching via Sequential Monte Carlo

Yumin Suh, Minsu Cho, Kyoung Mu Lee

Graph Matching via Sequential Monte Carlo

Yumin Suh[1], Minsu Cho[2], and Kyoung Mu Lee[1]

[1] Department of EECS, ASRI, Seoul National University, Seoul, Korea
[2] INRIA - WILLOW / École Normale Supérieure, Paris, France

Abstract. Graph matching is a powerful tool for computer vision and machine learning. In this paper, a novel approach to graph matching is developed based on the sequential Monte Carlo framework. By constructing a sequence of intermediate target distributions, the proposed algorithm sequentially performs a sampling and importance resampling to maximize the graph matching objective. Through the sequential sampling procedure, the algorithm effectively collects potential matches under one-to-one matching constraints to avoid the adverse effect of outliers and deformation. Experimental evaluations on synthetic graphs and real images demonstrate its higher robustness to deformation and outliers.

Keywords: graph matching, sequential Monte Carlo, feature correspondence, image matching, object recognition.

1 Introduction

Graphs provide excellent means to represent features and the structural relationship between them, and the problem of feature correspondence is effectively solved by graph matching. Therefore, graph matching has gained high popularity in various applications in computer vision and machine learning (e.g., object recognition [1], shape matching [2], and feature tracking [3]). Due to the NP-hard nature of graph matching in general, its optimal solution is virtually unachievable so that a myriad of algorithms are proposed to approximate it. In recent computer vision literature, the graph matching problem is widely formulated as the Integer Quadratic Program (IQP) [4,5,6,7] that explicitly considers both local appearances and generic pairwise relations. This line of graph matching researches has further extended to hyper-graph matching [8,9], learning techniques [10,11], and its progressive framework [12]. The existing methods, however, still suffer from severe distortion of the graphs (e.g., addition of outliers in nodes or deformation of attributes). Robustness to such distortion is of particular importance in practical vision problems because real-world image matching problems are widely exposed to background clutter and image variation.

In this paper, a robust graph matching algorithm based on the Sequential Monte Carlo (SMC) framework [13] is proposed. Unlike previous methods, the algorithm sequentially samples potential matches via proposal distributions to maximize the graph matching objective in a robust and effective manner. To treat the graph matching problem in the SMC framework, a sequence of artificial target distributions is designed, which provide smooth transitions from an

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part III, LNCS 7574, pp. 624–637, 2012.

initial distribution to the final target distribution as the graph matching objective. The proper design of an importance distribution and a resampling scheme allows the algorithm to explore the solution space efficiently under the matching constraints. Consequently, the proposed algorithm achieves higher robustness to outliers and deformation than other graph matching algorithms.

Among numerous graph matching methods, the algorithms closely related to ours are as follows. Maciel and Costeira [14] casted a graph matching problem as an integer constrained minimization, which is popularly adopted for its generality in recent studies including ours. Many of the subsequent approaches to the IQP formulation handle the original problem by solving a relaxed problem and then enforcing the matching constraint on its solution. For example, Gold and Rangarajan [15] relaxed the IQP by dropping the integer constraint and iteratively estimated the convex approximations around the previous solution. Leordeanu et al. [4] and Cour et al. [16] proposed spectral types of relaxation followed by a discretization process. Torresani et al. [17] used dual decomposition technique for graph matching optimization. Cho et al. [6] introduced a random walk algorithm to explore the relaxed solution space in the consideration of the matching constraints. More recently, to overcome the limitation of pair-wise relations and embed higher-order information, hyper-graph matching techniques were proposed based on the generalization of the IQP [8,9].

There have been many sampling-based algorithms for feature correspondence and shape matching in literature. Dellaert et al. [18] used the Metropolis-Hastings (MH) scheme to solve a correspondence problem in a Bayesian framework. Cao et al. [2] used the MH scheme for partial shape matching. Tamminen et al. [19] used the SMC framework to solve Bayesian matching of objects with occlusion. Lu et al. [20] and Yang et al. [21] proposed a shape detection algorithm based on the particle filter with static observations. However, all these methods for shape matching problems cannot be applied to graph matching in general. Lee et al. [7] introduced the data-driven Markov chain Monte Carlo (DDMCMC) algorithm for graph matching which adopts spectral relaxation for data-driven proposals. To our best knowledge, our work is the first attempt to use the SMC framework for the matching of general graphs.

2 Problem Definition

We consider two attributed graphs, $G^P = (\mathcal{V}^P, \mathcal{E}^P, \mathcal{A}^P)$ and $G^Q = (\mathcal{V}^Q, \mathcal{E}^Q, \mathcal{A}^Q)$, where $\mathcal{V}$, $\mathcal{E}$, and $\mathcal{A}$ denotes a set of nodes, edges, and attributes, respectively. In visual feature matching problems, attribute $\mathbf{a}_i^P \in \mathcal{A}^P$ of node $v_i^P \in \mathcal{V}^P$ usually describes a local appearance at feature i in image P, and attribute $\mathbf{a}_{ij}^P \in \mathcal{A}^P$ of edge $e_{ij}^P \in \mathcal{E}^P$ represents geometric relationship between features i and j in image P. In this paper, one-to-one matching constraints are adopted that every node in G^P is mapped to at most one node in G^Q and vice versa. Then, *Graph matching* is to find node correspondences between G^P and G^Q, which best preserve the attribute relations under the matching constraints. A graph matching solution $\mathcal{M}$ is represented by a set of *assignments* or *matches* (v_i^P, v_a^Q) between

nodes $v_i^P \in \mathcal{V}^P$ and $v_a^Q \in \mathcal{V}^Q$. $\mathcal{M}$ can be alternatively expressed by an *assignment matrix* $\mathbf{X} \in \{0,1\}^{n^P \times n^Q}$, where n^P and n^Q are the numbers of nodes in G^P and G^Q, respectively; $\mathbf{X}_{i,a} = 1$ if $(v_i^P, v_a^Q) \in \mathcal{M}$ and zero otherwise. An *assignment vector* $\mathbf{x}$ denotes the column-wise concatenated vector of $\mathbf{X}$, and the element of $\mathbf{x}$ corresponding to $\mathbf{X}_{i,a}$ is represented by $\mathbf{x}(i,a)$. The *affinity matrix* $\mathbf{W}$ consists of the relational similarity values between edges and nodes; the compatibility of two edge attributes $\mathbf{a}_{ij}^P$ and $\mathbf{a}_{ab}^Q$, or equivalently two matches (v_i^P, v_a^Q) and (v_j^P, v_b^Q), is encoded in non-diagonal component $\mathbf{W}_{i,a;j,b}$, and the compatibility of two node attributes $\mathbf{a}_i^P$ and $\mathbf{a}_a^Q$ is encoded in the diagonal component $\mathbf{W}_{i,a;i,a}$. The graph matching score of $\mathbf{x}$ is evaluated by $\mathbf{x}^T\mathbf{W}\mathbf{x}$ which corresponds to the sum of all similarity values covered by the matching. Therefore, graph matching is mathematically formulated as the following IQP problem [4,7,6]:

$$\begin{aligned} \mathbf{x}^* &= \operatorname{argmax}(\mathbf{x}^T\mathbf{W}\mathbf{x}) \\ s.t. \quad & \mathbf{x} \in \{0,1\}^{n^P n^Q}, \\ \forall i,\ \textstyle\sum_a \mathbf{x}(i,a) \le 1, \ \ & \forall a,\ \textstyle\sum_i \mathbf{x}(i,a) \le 1. \end{aligned} \tag{1}$$

3 Sequential Monte Carlo for Graph Matching

In this paper, graph matching is formulated as a sequential sampling of potential matches based on the SMC framework. The SMC methods (e.g., the particle filter [22]) are widely used for dynamic problems in computer vision, such as object tracking and structure from motion. The basic concept of the SMC is to sequentially estimate the target probability distribution π_t at time t with a set of *particles*, that is a cloud of weighted random samples $\{(\mathbf{x}_t^{(i)}, w_t^{(i)})\}_{i=1}^N$. A weight $w_t^{(i)}$ represents the importance of the associated sample $\mathbf{x}_t^{(i)}$ [23].

The samples $\mathbf{x}_t^{(i)}$ at time t are generated from the importance distribution η_t, which is constructed by a Markov transition kernel $\mathcal{K}$ and the previous distribution π_{t-1} [24]; each sample $\mathbf{x}_t^{(i)}$ evolves from $\mathbf{x}_{t-1}^{(i)}$ according to the kernel $\mathcal{K}$, and the importance distribution η_t is represented by

$$\eta_t(\mathbf{x}_t) = \sum_{\mathbf{x}_{t-1}} \pi_{t-1}(\mathbf{x}_{t-1})\mathcal{K}(\mathbf{x}_{t-1}, \mathbf{x}_t). \tag{2}$$

The weight $w_t^{(i)}$ of each sample $\mathbf{x}_t^{(i)}$ is proportional to the ratio of the target distribution π_t to the importance distribution η_t, and is defined as

$$W_t^{(i)} = \frac{\pi_t(\mathbf{x}_t^{(i)})}{\eta_t(\mathbf{x}_t^{(i)})}, \quad w_t^{(i)} = \frac{W_t^{(i)}}{\sum_{i=1}^N W_t^{(i)}}. \tag{3}$$

Then, the *particles* as the *weighted samples* asymptotically converges to the target distribution:

$$\tilde{\pi}_t(\mathbf{x}_t) = \sum_{i=1}^{N} w_t^{(i)} \delta_{\mathbf{x}_t^{(i)}}(\mathbf{x}_t) \xrightarrow{a.s.} \pi_t(\mathbf{x}_t) \quad \text{as} \quad N \to \infty, \tag{4}$$

where δ denotes the Dirac delta function.

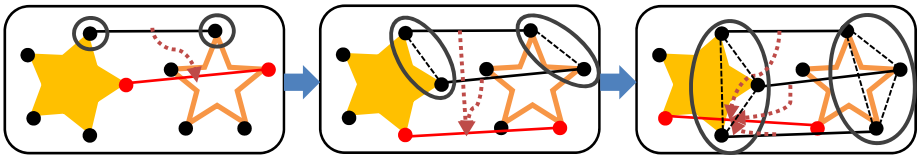


Fig. 1. The main idea of the proposed algorithm. The algorithm sequentially samples potential matches via proposal distributions to maximize the graph matching objective under the matching constraints. In every time step, new matches are sampled based on the affinity with the previously sampled matches.

Unlike typical SMC methods for dynamic problems [22], we do not assume any prior knowledge about the graph for generality, except for the affinity matrix. In this case, it is intractable to calculate the conditional distribution even up to a normalizing constant, thus the direct sequential sampling from conditional distributions is also intractable.

To formulate graph matching in the SMC framework, we introduced a sequence of intermediate distributions [24], which starts from a simple initial distribution and gradually moves toward the final target distribution equivalent to the objective function of (1). This SMC procedure performs graph matching by sequentially re-distributing particles over the evolving sample space according to the intermediate target distribution at each step. The procedure is illustrated in Fig. 2, and the details are described in the following subsections.

3.1 Sequential Target Distribution

In the SMC graph matching, each particle incrementally collects a new match at each time step so that each particle $\mathbf{x}_t^{(i)}$ at time step t always has t matches. The maximum number of matches in the solution (or the maximum time step in the sampling) is $L = \min(n^P, n^Q)$. We design the final target distribution as

$$\pi_L(\mathbf{x}_L) \propto \exp(\mathbf{x}_L^T \mathbf{W} \mathbf{x}_L / \tau) \tag{5}$$

where τ is a scaling constant. Since the exponential function increases monotonically, the optimal solution of the graph matching score $\mathbf{x}^T\mathbf{W}\mathbf{x}$ in Eq. (1) maximizes the probability π_L of Eq. (5). Since $\mathbf{x}_L$ has high dimensionality of $n^P n^Q$, direct sampling from π_L is not tractable in general cases. As illustrated in Fig. 2(a), a sequence of intermediate target distributions $\{\pi_t\}_{t=1}^{L-1}$ is introduced for the SMC sampling as follows:

$$\pi_t(\mathbf{x}_t) \propto \exp(\mathbf{x}_t^T \mathbf{W} \mathbf{x}_t / \tau), \quad t = 1, \cdots, L-1. \tag{6}$$

A feasible set of the graph matching problem, or a set of all possible $\mathbf{x}$ satisfying the constraints of Eq. (1), is denoted by $\mathcal{X}$. Each intermediate target distribution π_t in Eq. (6) then is defined on the measurable space $(\mathcal{X}_t, \epsilon_t)$ where $\mathcal{X}_t = \{\mathbf{x} \in \mathcal{X} | \sum_{i,a} \mathbf{x}_t(i,a) = t\}$. The more true matches a particle $\mathbf{x}_t^{(i)}$ includes, the higher

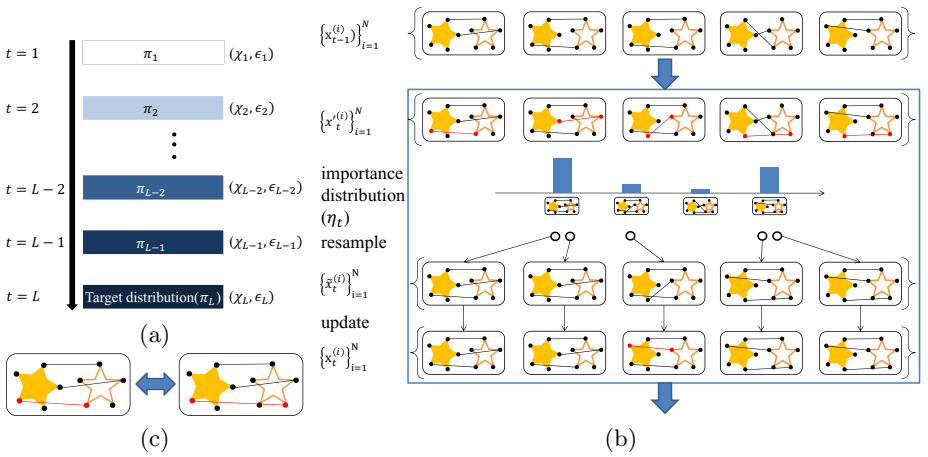


Fig. 2. Overview of the proposed method. (a) A sequence of intermediate target distributions from a simple initial distribution to the target distribution is constructed. (b) Transition from time step $t-1$ to the t is performed based on the sampling and importance resampling (SIR) technique exploiting the particles from the previous distribution. (c) An example of the neighborhood relation between two candidate matchings. They share two common matches and differ in one match.

value the probability $\pi_t(\mathbf{x}_t^{(i)})$ has. Thus, the sequence of distributions $\{\pi_t\}_{t=1}^{L-1}$ effectively drives the particles to the final target distribution π_L.

3.2 Sequential Sampling

In order to sample from the intermediate target distribution π_t, new particles are propagated from the previous particles $\{\mathbf{x}_{t-1}^{(i)}\}_{i=1}^{N}$ using the sampling and importance resampling (SIR) technique [22]. In each time step, each particle additionally selects a new match m in a probabilistic manner according to the transition kernel,

$$\mathcal{K}(\mathbf{x}_{t-1}, \mathbf{x}_{t-1}+\mathbf{e}_m) = \begin{cases} \frac{1}{Z}\exp(\mathbf{e}_m^T \mathbf{W} \mathbf{x}_{t-1}/\tau) & \text{if } (\mathbf{x}_{t-1}+\mathbf{e}_m) \in \mathcal{X}_t \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $\mathbf{e}_m$ is a unit vector that corresponds to the match m and Z is a normalizing constant. In selecting the new match m, the kernel reflects both the matching constraints and the affinity relations with previously sampled matches in $\mathbf{x}_{t-1}$. As the time step increases, previously selected inlier matches boost the sampling probability of other inlier matches for each particle. The augmented particle in this step is denoted by $\mathbf{x}'_t = \mathbf{x}_{t-1} + \mathbf{e}_m$. Given that particles at time step $t-1$ follow the distribution π_{t-1}, the transition leads them to a new importance distribution

$$\eta_t(\mathbf{x}'_t) = \sum_{\mathbf{x}_{t-1}} \pi_{t-1}(\mathbf{x}_{t-1})\mathcal{K}(\mathbf{x}_{t-1}, \mathbf{x}'_t) \approx \sum_{i=1}^{N} \pi_{t-1}(\mathbf{x}_{t-1}^{(i)})\mathcal{K}(\mathbf{x}_{t-1}^{(i)}, \mathbf{x}'_t). \quad (8)$$

For further discussion, let us denote the particles after transition by $\{\mathbf{x'}_t^{(i)}\}_{i=1}^N$. The particles of equal matches are grouped together to form a set of n unique assignment vectors $\{\mathbf{g'}_t^{(j)}\}_{j=1}^n$ and the number of particles corresponding to the j-th group is denoted by n_j. Then, the weight of the particle is approximated as follows:

$$W_t^{(i)} = \frac{\pi_t(\mathbf{x'}_t^{(i)})}{\eta_t(\mathbf{x'}_t^{(i)})} \approx \frac{\pi_t(\mathbf{x'}_t^{(i)})}{\frac{1}{N}\sum_{j=1}^{n} n_j \delta(\mathbf{x'}_t^{(i)} - \mathbf{g'}_t^{(j)})} \propto \frac{\pi_t(\mathbf{x'}_t^{(i)})}{n_{I(i)}} \quad (9)$$

where $I(i)$ is a group ID of the particle $\mathbf{x'}_t^{(i)}$. Therefore, to calculate the weights, particles of same matches should be grouped together first as detailed in Sec. 3.4. Then, the weights are easily obtained by computing $\pi_t(\mathbf{x}_t)$. Finally, the resampled particles based on the weights, $\{\tilde{x}_t^{(i)}\}_{i=1}^N$, follow the intermediate target distribution π_t.

3.3 Particle Updating

To correct false matches sampled in early steps, we update particles after sampling at every time step. As shown in Fig. 2(c), the particle updating spreads the current particle to a neighborhood support sharing $t-1$ matches by changing the most unreliable match to a better match. Among the matches in the particle $\tilde{\mathbf{x}}_t^{(i)}$, the match sharing the lowest affinity values with other matches is selected by

$$m_w = \underset{m \ s.t. \ \mathbf{e}_m^T \tilde{\mathbf{x}}_t^{(i)} = 1}{\operatorname{argmin}} \mathbf{e}_m^T \mathbf{W} \tilde{\mathbf{x}}_t^{(i)}. \quad (10)$$

The selected match m_w is replaced with a new match sampled by the transition kernel $\mathcal{K}(\tilde{\mathbf{x}}_t^{(i)} - \mathbf{e}_{m_w}, \mathbf{x}_t^{(i)})$ of Eq. (7). After repeating the grouping and resampling process of Sec. 3.2, the updated particles $\{\mathbf{x}_t^{(i)}\}_{i=1}^N$ are obtained. In this updating procedure, while following the intermediate target distribution, the particles are redistributed into the support with high probability under the target distribution.

3.4 Implementation Details

The proposed algorithm is summarized in Algorithm 1. Some implementation issues are addressed in the following.

Computational Complexity. One step procedure is composed of transition, grouping, resampling, and updating. The computational complexity for each subprocedure is as follows: Transition $O(Nn_{cm} + G_t n_{cm} + G_t t)$, grouping $O(Nt)$ (n_{cm} : the number of candidate matches, N : the number of particles, G_t : the number of unique groups at time step t, $G_t \leq N$, $t \leq L \leq n_{cm}$). Therefore the

Algorithm 1: SMC graph matching

input : affinity matrix $\mathbf{W}$
output: assignment vector $\mathbf{x}$

// Initialization
Sample particles from the initial distribution π_1;
// SMC loop
while *sampling available* **do**
  Propagate each particle by sampling a new match according to Eq. (7);
  Resample the particles according to the weights of Eq. (9);
  Update the particles according to Sec. 3.3;
end
Choose a particle $\mathbf{x}^*$ with the highest matching score as the final solution;

total complexity for one step procedure is $O(Nn_{cm})$, and the entire algorithm takes $\mathcal{O}(n_{cm}LN)$ where L denotes the number of samplings.

Initial Distribution. The initial distribution is calculated as the marginal of the intermediate target distribution, π_2. Utilizing the symmetric property of the affinity matrix, the initial distribution π_1 is obtained as

$$\pi_1(\mathbf{x}_1) \propto \mathbf{1}^T \exp(\mathbf{W}/\tau)\mathbf{x}_1, \tag{11}$$

where $\exp(\mathbf{W}/\tau)$ denotes the elementwise exponent of $(\mathbf{W}/\tau)$ and $\mathbf{1}$ is a column vector with all ones.

Transition Kernel. In order to reduce the computational cost and memory cost, a transition kernel $\mathcal{K}$ is computed as an approximated version which its lowest $100(1-\alpha)\%$ was forced to be 0. Experimental results for different α are shown in synthetic random graph matching task in Sec. 4.3.

Grouping. Grouping of particles can be performed efficiently by using radix sort and a simple trick. First, a unique ID is assigned to every candidate match. Then each particle is represented by a combination of IDs. IDs in each combination are sorted so that the sequence of IDs form a N-ary t digit number with the i-th digit value corresponding to the i-th smallest ID. Now, after N-ary radix sort, particles can be efficiently grouped in $\mathcal{O}(Nt)$ (N:the number of particles, t: match size) by comparing only the adjacent neighbors.

Final Solution. The SMC loop is terminated simply when no more sampling is possible while keeping the constraints. Then, a particle maximizing the score of Eq. (1) is selected as the final solution. Since the matches of each particle already satisfy the one-to-one matching constraints, no additional post-processing for discretization is needed in our approach.

4 Experiments

In this section, two sets of experiments were performed for quantitative evaluation: (1) synthetically generated random graph matching and (2) feature matching using real images. The proposed algorithm (SMCM) was compared with

seven state-of-the-art methods including SM[4], SMAC[16], PM[25], IPFP[5], GAGM[15], RRWM[6], and DDMCM[7]. The authors' implementations were used for SM, SMAC[1], PM[2], IPFP[3], DDMCM, and RRWM[4]. For GAGM, the implementation by Cour et al. [16] is adopted. All methods were implemented using MATLAB and were tested on 2.40 GHz Core2 Quad desktop PC. In each experiment, the same affinity matrix was shared by all algorithms, and α is fixed to 0.1 in SMCM. In addition to the two experiments, the experimental analyses of SMCM will be provided in Sec. 4.3.

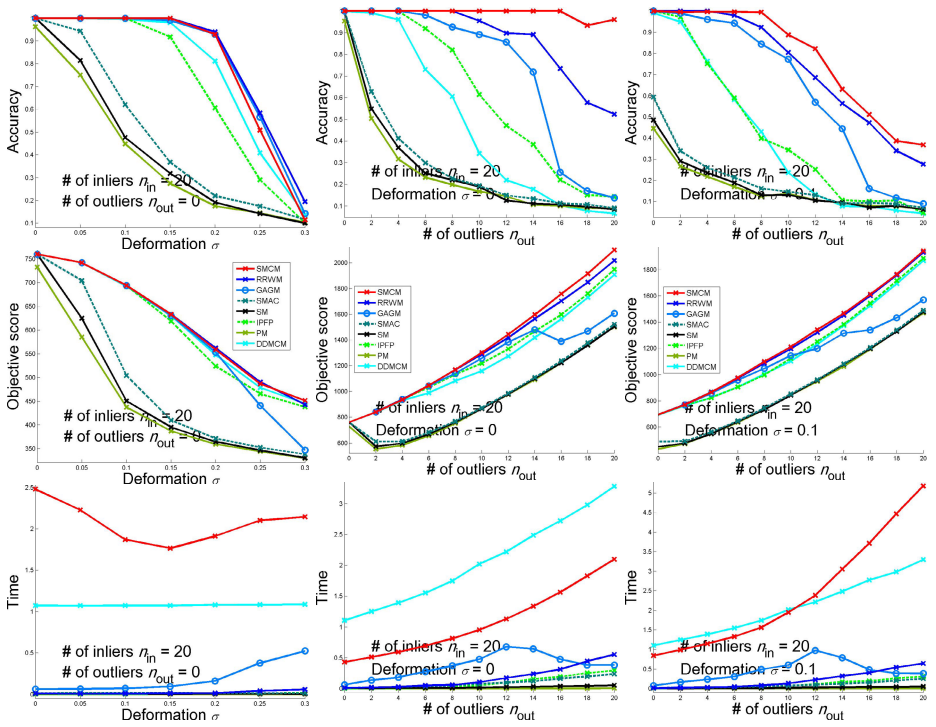
4.1 Synthetic Random Graph Matching

The random graph matching experiments were performed following the experimental protocol of [16,15,6]. In this experiment, the Hungarian algorithm[26] was used for the post-processing of discretization, and $\tau = 2$ is used for SMCM.

Two graphs, G^P and G^Q, were constructed at each trial with $n = n_{in} + n_{out}$ nodes, where n_{in} and n_{out} denote number of inlier and outlier nodes, respectively. The reference graph G^P was generated with random edges, where each edge $(i,j) \in \mathcal{E}^P$ was assigned a random attribute $\mathbf{a}^P_{ij}$ distributed uniformly in $[0,1]$. A perturbed graph G^Q was then created by adding noise on the edge attributes between inlier nodes: $\mathbf{a}^Q_{p(i)p(j)} = \mathbf{a}^P_{ij} + \varepsilon$, where $p(.)$ is a random permutation function. The deformation noise ε was assigned by the Gaussian function $\mathcal{N}(0, \sigma^2)$. All the other edges connecting at least one of the outlier nodes were randomly generated with random edges in the same way of G^P. Thus, G^P and G^Q share a common but perturbed subgraph with size n_{in}. The affinity matrix $\mathbf{W}$ was computed by $\mathbf{W}_{i,a;j,b} = \exp(-|\mathbf{a}^P_{i,j} - \mathbf{a}^Q_{a,b}|^2/\sigma_s^2)$, $\forall e^P_{ij} \in \mathcal{E}^P, \forall e^Q_{ab} \in \mathcal{E}^Q$ and the diagonal terms are set to be zero. The scaling factor σ_s^2 was chosen to be 0.1 to perform best in overall algorithms. The accuracy was computed as the ratio of the number of true positive matches to that of ground truths. The objective score was computed by the IQP objective $\mathbf{x}^T\mathbf{W}\mathbf{x}$.

In the experimental setting, there are two independent variables, the number of outliers n_{out} and the amount of deformation noise σ. Using these, three experiments with different conditions were conducted: (1) varying the amount of deformation noise with no outliers (2) varying the number of outliers with no deformation (3) varying the number of outliers with some deformation. Each experiment was repeated for 100 times with different matching problems generated from the above procedure, and the average accuracy and objective score were recorded. In all experiments, the number of inliers n_{in} is fixed to 20. Figure 3(a) represents the results of varying deformation noise σ from 0 to 0.3 by increments of 0.05 while fixing the number of outliers $n_{out} = 0$ and particles $N = 10000$. Figure 3(b) shows the result of varying the number of outliers n_{out} from 0 to 20 by increments of two while fixing $N = 2000$ and deformation noise $\sigma = 0$.

[1] http://www.seas.upenn.edu/~timothee/
[2] http://www.cs.huji.ac.il/~zass/
[3] http://86.34.14.245/code.php
[4] http://cv.snu.ac.kr/research/~RRWM/



(a) deformation (b) outliers (c) outliers with deform

Fig. 3. Synthetic graph matching experiments. The accuracies (top), objective scores (middle) and computational times (bottom) are plotted for three different conditions: (a) varying deformation noise with no outliers, (b) varying the number of outliers with no deformation noise, and (c) varying the number of outliers with some deformation noise.

Figure 3(c) represents the result of varying the number of outliers n_{out} from 0 to 20 by increments of two while fixing $N = 5000$ and deformation noise $\sigma = 0.1$.

As observed in the score plots (middle) in Fig. 3, SMCM consistently outperforms all the other algorithms in the objective score, regardless of the degree of deformation and outliers; SMCM performs best as the optimizer of graph matching score. Especially, the accuracy of SMCM significantly exceeds those of the other algorithms under increasing outliers as shown in Figs. 3(b) and 3(c). It implies that in the SMC framework SMCM avoids the adverse effects of outliers effectively through sequential collection of inliers under the matching constraints. In deformation experiment of Fig. 3(a), the accuracy of SMCM is comparable or slightly inferior to the best of RRWM. However, SMCM outperforms all other methods in realistic conditions where both outliers and deformation exists as in Fig. 3(c). The bottom row of Fig. 3 shows the computational time.

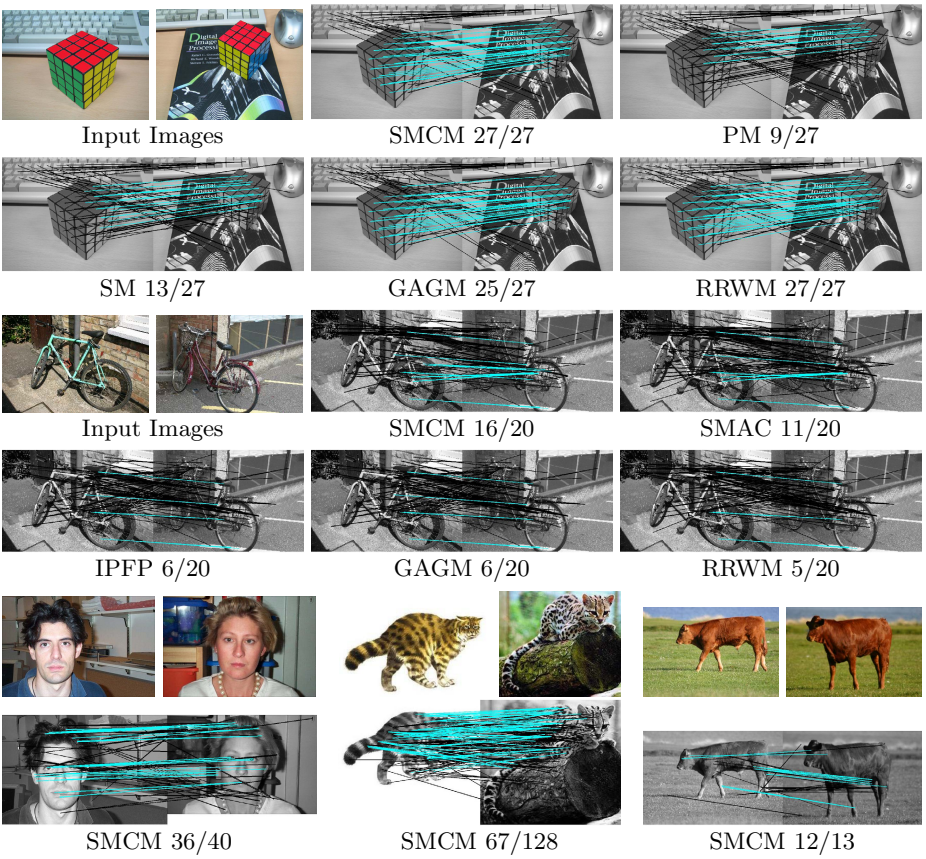


Fig. 4. Some representative results of the real image matching experiments. True positive matches are represented by blue lines and false positive matches are represented by black lines.

4.2 Real Image Matching

In this experiment, we performed real image matching on the benchmark image dataset[5] used in [6]. The dataset consists of 30 image pairs containing photos of various objects, most of which are collected from Caltech-101 and MSRC datasets, and provides the detected MSER features [27], the initial matches, and the manually-labeled ground truth matches. For each image pair, two graphs were constructed by considering the features as nodes. To sparsify the affinity matrix, only the similarity values related to candidate correspondences are assigned in the affinity matrix. In our experiments, 70 nodes and 720 candidate matches per an image pair are used in average. The affinity matrix $\mathbf{W}_{i,a;j,b} = \exp(-d_{i,a;j,b}/\sigma_s^2)$ is computed using the symmetric transfer error

[5] The dataset is available from `http://cv.snu.ac.kr/research/~RRWM/`

Table 1. Avergage performance on the real images

Algorithm	SM	RRWM	PM	IPFP	SMAC	GAGM	SMCM
Accuracy (%)	65.3	69.0	60.5	65.7	50.8	67.6	**69.6**
Time (Sec.)	0.02	1.00	0.05	0.13	0.03	1.67	0.89

$d_{i,a;j,b}$, which is used in [6], as the measure of affine-invariant dissimilarity between two feature pairs, (v_i^P, v_j^P) and (v_a^Q, v_b^Q). The scaling parameter σ_s^2 was set to 10 to enhance the overall accuracy over all algorithms. In this experiment, the greedy mapping was used for the post-processing of discretization, and $\tau = 10$ and 500 particles are used for SMCM. The matching accuracy and the running time were measured for SM, RRWM, PM, IPFP, SMAC, GAGM, and SMCM. [6] The results are summarized in the Table 1 and some examples are shown in Fig. 4. SMCM records the highest accuracy in acceptable running time. Its robustness to outliers and deformations is observed again in this real image matching experiment.

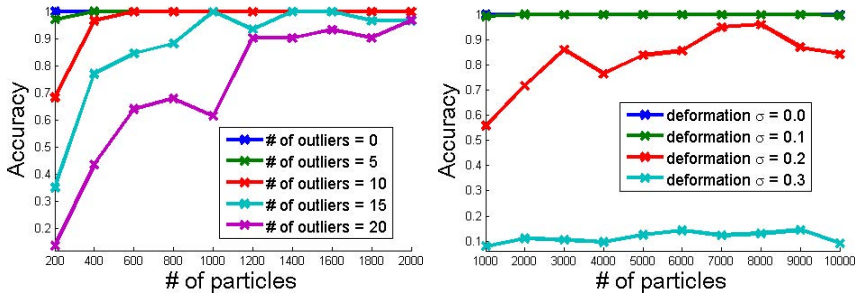


Fig. 5. To determine the proper number of particles experimentally, we plotted the average accuracy over 30 experiments with varying the number of particles and problem difficulty in the synthetic random graph matching experiment (Sec. 4.1). The plots are obtained by varying (a) the number of outliers and (b) deformation rate. In both cases, the accuracy saturates as the number of particles become large enough, except for the extreme conditions.

4.3 More Experimental Analyses

In this section, we performed three additional experiments to analyze the characteristics of SMCM. In the first experiment, we evaluated the performance of SMCM while varying number of particles. The larger the number of particles is, the more likely the set of matches with low probability are sampled, resulting

[6] For SMCM, we measured the average accuracy over 30 trials because SMCM is a stochastic algorithm unlike others.

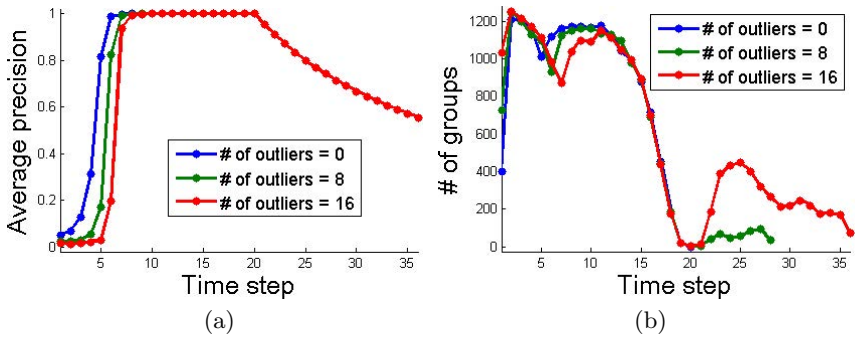


Fig. 6. The average precision of particles and the number of particle groups at each time step are measured on the synthetic random graph matching in Sec. 4.1. (a) The average precision of particles increases as the number of matches in particles increases. The plot is depicted while the number of outliers is varied. An abrupt increase occurs when enough true matches (3–8) are sampled. (b) The number of particle groups is depicted versus the time step while the number of outliers is varied. The number of groups decreases as the match size increases.

in broader search. Therefore, more particles produce better performance at the cost of more computation. There is a trade-off between accuracy and time, and it can be controlled by the number of particles, depending on the application. Using the same synthetic random graph experiment of Sec. 4.1, we plotted the average accuracy over 30 experiments with respect to the number of particles. Figures 5(a) and 5(b) provide the results varying the number of outliers with no deformation, and varying deformation noise with no outliers, respectively. For the problem with outliers, 400–1000 samples are needed for saturation, and in the presence of deformation, more particles, about 5000, are needed to arrive at saturation point. Unlike outlier noise, the deformation noise lowers the affinity value between true matches, making true and false matches indistinguishable. Therefore, more particles are needed in the presence of deformation noise. Nevertheless, the required number of particles is only a tiny fraction of the number of all possible combinations (20!).

In the second experiment, we evaluated the efficiency of the algorithm. First, the average precision of particles at each time step is depicted in Fig. 6(a) for the synthetic random graph matching experiment in Sec. 4.1 while the number of outliers is varied. It increases abruptly after about three to eight samplings, which means that previously sampled matches in the particles boost sampling of other inlier matches. It decreases after time step 20 because the number of true matches is 20. Secondly, Fig. 6(b) shows the number of groups at each time step in the same experiments. Since the information of particles is shared within a group, the number of groups directly affects the computational cost. In Fig. 6, it decreases until time step 3–8 while average precision increases, which implies that the weights are highly concentrated on reliable groups at the early stages.

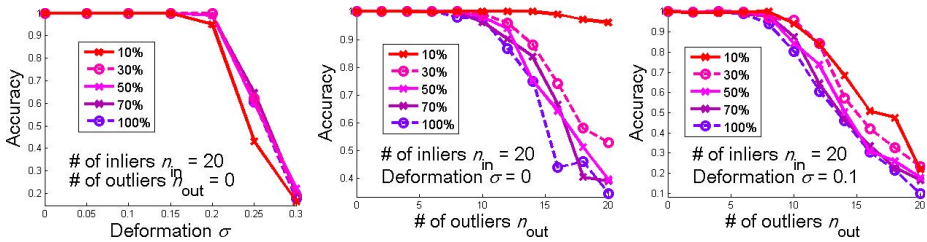


Fig. 7. The accuracies for different settings of α are shown. The experimental settings are same as the synthetic random graph matching in Sec. 4.1.

When the precision saturates, the number of groups gradually decreases after a small increase, resulting in effective reduction of computational cost.

In the third experiment, we evaluated the accuracy for different settings of α for the same experimental settings in Sec. 4.1. The results are shown in Fig. 7. As the value of α increases, SMCM becomes more robust to the deformation noise. In the presence of outliers, however, the highest accuracy is achieved at $\alpha = 0.1$ where the distracting outlier matches with relatively small affinity values are effectively ignored.

5 Conclusion

In this paper, we proposed a novel graph matching algorithm based on the Sequential Monte Carlo (SMC) framework. During the sequential sampling procedure, our algorithm effectively collects potential matches under matching constraints and avoids the adverse effect of outliers and deformation. The experimental results show that it outperforms the state-of-the-art algorithms and is highly robust to outliers. Since the SMC can preserves samples of multi-modes in particles, it is extendable to multi-modal graph matching algorithm by utilizing the properties. We leave this in future work.

References

1. Duchenne, O., Joulin, A., Ponce, J.: A graph-matching kernel for object categorization. In: ICCV (2011)
2. Cao, Y., Zhang, Z., Czogiel, I., Dryden, I., Wang, S.: 2d nonrigid partial shape matching using mcmc and contour subdivision. In: CVPR (2011)
3. Birchfield, S.: KLT: An implementation of the kanade-lucas-tomasi feature tracker (1998)
4. Leordeanu, M., Hebert, M.: A spectral technique for correspondence problems using pairwise constraints. In: ICCV (2005)
5. Leordeanu, M., Hebert, M.: An integer projected fixed point method for graph matching and map inference. In: NIPS (2009)

6. Cho, M., Lee, J., Lee, K.M.: Reweighted Random Walks for Graph Matching. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 492–505. Springer, Heidelberg (2010)
7. Lee, J., Cho, M., Lee, K.M.: A graph matching algorithm using data-driven markov chain monte carlo sampling. In: ICPR (2010)
8. Duchenne, O., Bach, F., Kweon, I.S., Ponce, J.: A tensor-based algorithm for high-order graph matching. PAMI (2010)
9. Lee, J., Cho, M., Lee, K.M.: Hyper-graph matching via reweighted random walks. In: CVPR (2011)
10. Caetano, T., McAuley, J., Cheng, L., Le, Q., Smola, A.: Learning graph matching. PAMI (2009)
11. Leordeanu, M., Hebert, M.: Unsupervised learning for graph matching. In: CVPR (2009)
12. Cho, M., Lee, K.M.: Progressive graph matching: Making a move of graphs via probabilistic voting. In: CVPR (2012)
13. Cappé, O., Godsill, S., Moulines, E.: An overview of existing methods and recent advances in sequential monte carlo. Proc. IEEE (2007)
14. Maciel, J., Costeira, J.P.: A global solution to sparse correspondence problems. PAMI 25, 187–199 (2003)
15. Gold, S., Rangarajan, A.: A graduated assignment algorithm for graph matching. PAMI, 411–436 (1996)
16. Cour, T., Srinivasan, P., Shi, J.: Balanced graph matching. In: NIPS (2006)
17. Torresani, L., Kolmogorov, V., Rother, C.: Feature Correspondence Via Graph Matching: Models and Global Optimization. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 596–609. Springer, Heidelberg (2008)
18. Dellaert, F., Seitz, S.M., Thrun, S., Thorpe, C.: Feature correspondence: A markov chain monte carlo approach. In: NIPS (2001)
19. Tamminen, T., Lampinen, J.: Sequential monte carlo for bayesian matching of objects with occlusions. PAMI (2006)
20. Lu, C., Latecki, L.J., Adluru, N., Yang, X., Ling, H.: Shape guided contour grouping with particle filters. In: ICCV (2009)
21. Yang, X., Latecki, L.J.: Weakly Supervised Shape Based Object Detection with Particle Filter. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 757–770. Springer, Heidelberg (2010)
22. Isard, M., Blake, A.: Condensation - conditional density propagation for visual tracking. IJCV (1998)
23. Moral, P.D., Doucet, A.: Sequential monte carlo for bayesian computation. Journal of the Royal Statistical Society (2007)
24. Moral, P.D., Doucet, A., Jarsa, A.: Sequential monte carlo samplers. Journal of the Royal Statistical Society, 411–436 (2006)
25. Zass, R., Shashua, A.: Probabilistic graph and hypergraph matching. In: CVPR (2008)
26. Munkres, J.: Algorithms for the assignment and transportation problems. SIAM (1957)
27. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide baseline stereo from maximally stable extremal regions. In: BMVC (2002)